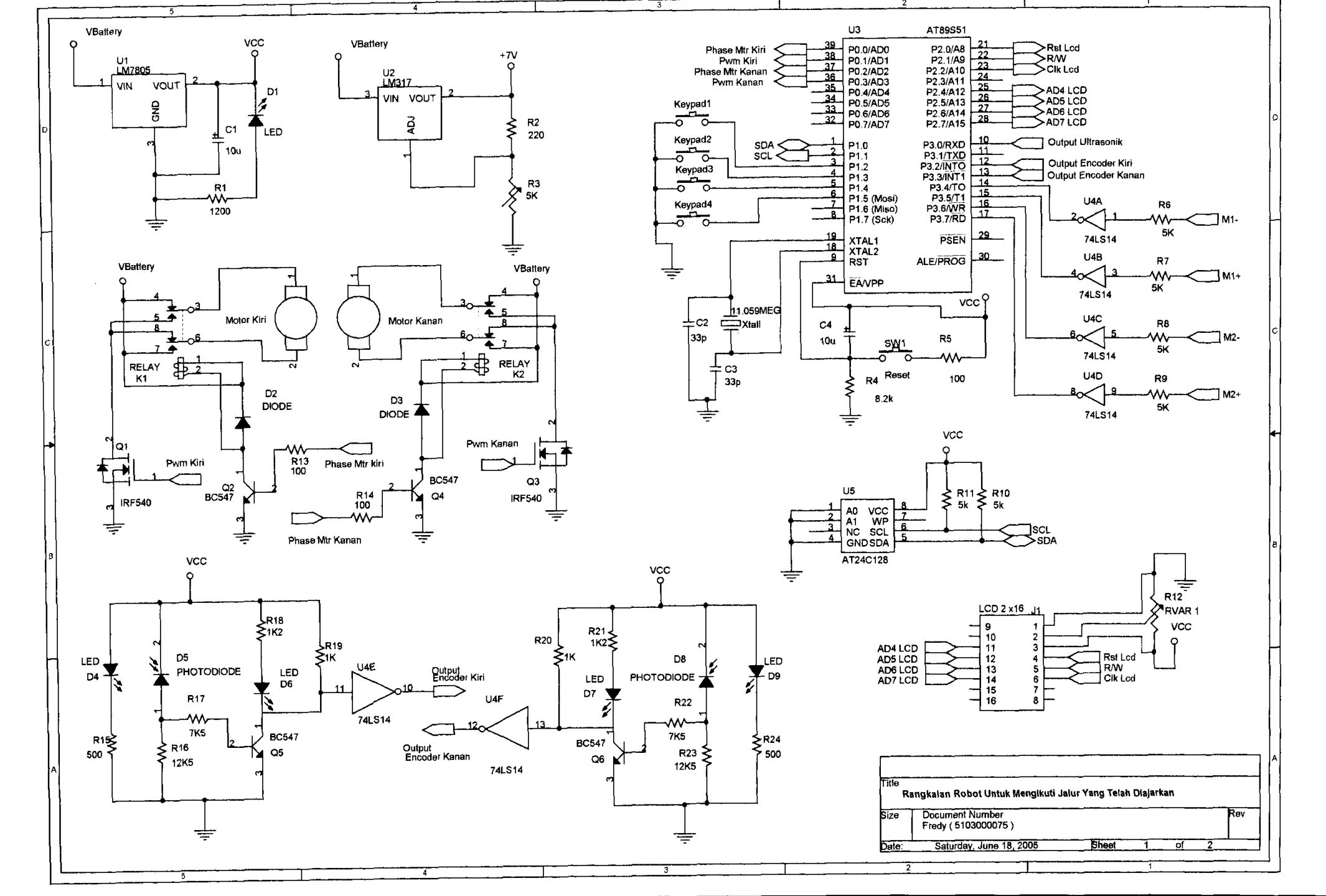
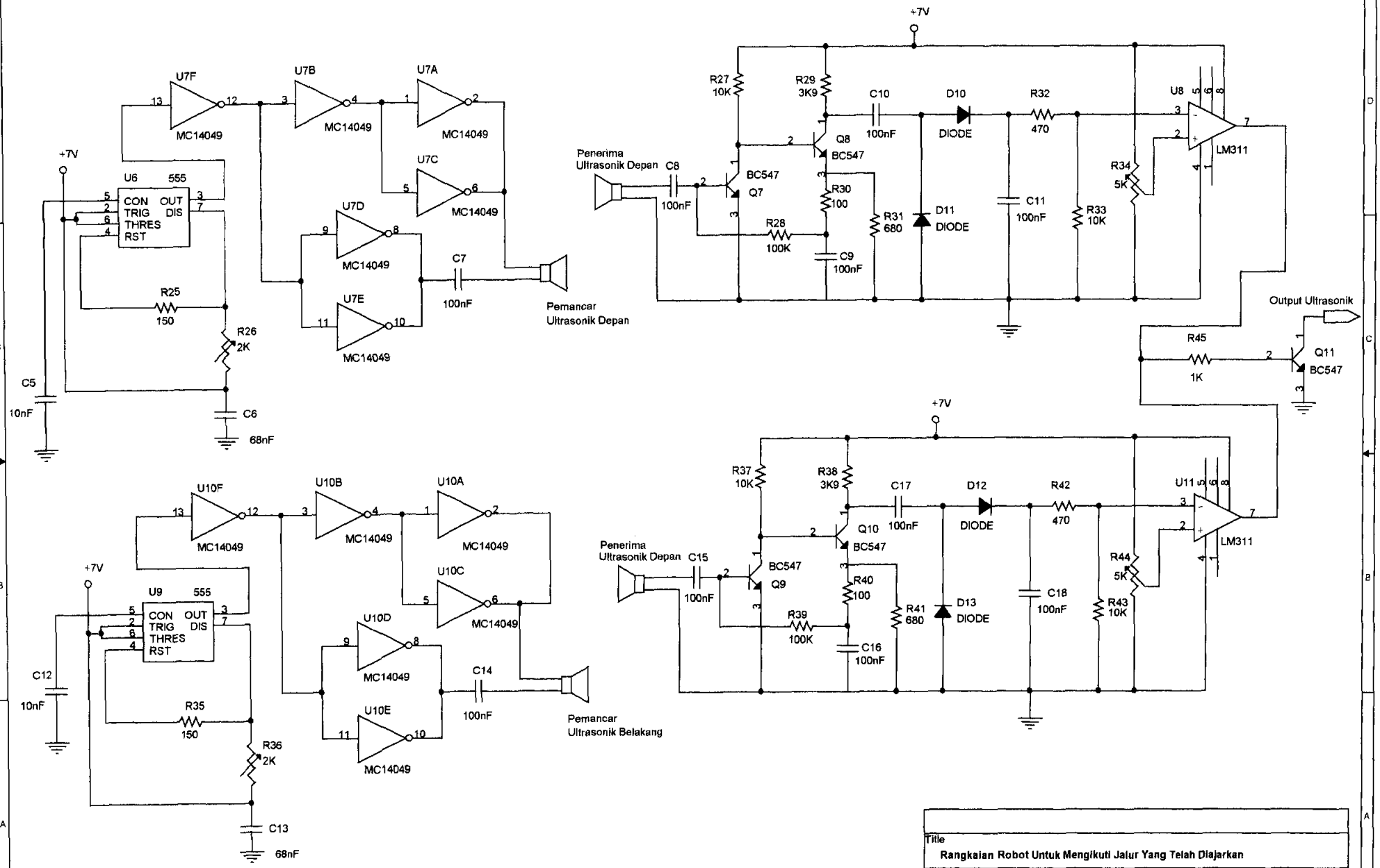
LAMPIRAN


VBattery
U1
LM7805
VIN
VOUT
GND
VCC
D1
LED
C1
10u
R1
1200
U2
LM317
ADJ
+7V
R2
220
R3
5K
U3
AT89S51
Phase Mtr Kiri
Pwm Kiri
Phase Mtr Kanan
Pwm Kanan
Keypad1
Keypad2
Keypad3
Keypad4
SDA
SCL
P0.0/AD0
P0.1/AD1
P0.2/AD2
P0.3/AD3
P0.4/AD4
P0.5/AD5
P0.6/AD6
P0.7/AD7
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5 (Mosi)
P1.6 (Miso)
P1.7 (Sck)
XTAL1
XTAL2
RST
EA/VPP
ALE/PROG
PSEN
P2.0/A8
P2.1/A9
P2.2/A10
P2.3/A11
P2.4/A12
P2.5/A13
P2.6/A14
P2.7/A15
P3.0/RXD
P3.1/TXD
P3.2/INT0
P3.3/INT1
P3.4/T0
P3.5/T1
P3.6/WR
P3.7/RD
Rst Lcd
R/W
Clk Lcd
AD4 LCD
AD5 LCD
AD6 LCD
AD7 LCD
Output Ultrasonik
Output Encoder Kiri
Output Encoder Kanan
C2
33p
11.059MEG
Xtall
C3
33p
C4
10u
SW1
Reset
R4
8.2k
R5
100
U4A
U4B
U4C
U4D
74LS14
R6
R7
R8
R9
5K
M1-
M1+
M2-
M2+
VBattery
Motor Kiri
Motor Kanan
RELAY
K1
K2
D2
DIODE
D3
DIODE
Q1
IRF540
Pwm Kiri
Q2
BC547
R13
100
Phase Mtr kiri
R14
100
Phase Mtr Kanan
BC547
Q4
Q3
IRF540
Pwm Kanan
U5
AT24C128
A0
A1
NC
GND
VCC
WP
SCL
SDA
R11
5k
R10
5k
LCD 2 x16
J1
R12
RVAR 1
LED
D4
PHOTODIODE
D5
LED
D6
R17
7K5
R15
500
R16
12K5
BC547
Q5
R18
1K2
R19
1K
U4E
74LS14
Output
Encoder Kiri
U4F
Output
Encoder Kanan
74LS14
R20
1K
R21
1K2
LED
D7
PHOTODIODE
D8
LED
D9
R22
7K5
R23
12K5
R24
500
BC547
Q6
Title
Rangkaian Robot Untuk Mengikuti Jalur Yang Telah Diajarkan
Size
Document Number
Fredy (5103000075)
Rev
Date:
Saturday, June 18, 2005
Sheet
1
of
2



Title	Rangkaian Robot Untuk Mengikuti Jalur Yang Telah Diajarkan	
Size	Document Number Fredy (5103000075)	Rev
Date: Saturday, June 18, 2005	Sheet 2 of 2	

Listing Program :

```
        org     00h
        ljmp    start
        org     003h
        ljmp    int_kiri
        org     00bh
        acall   pwm_kiri
        reti
        org     013h
        ljmp    int_kanan
        org     01bh
        acall   pwm_kanan
        reti

        dseg
        org     0010h
        pwm1:           ds      1
        pwm2:           ds      1
        arah_kiri:              ds      1
        arah_kanan:             ds      1
        lcd_brs1:               ds      1
        lcd_brs2:               ds      1
        posisi_psn:             ds      1
        data_lcd:               ds      1
        almt_data_lcd:          ds      1
        almt_mem_msb:           ds      1
        almt_mem_lsb:           ds      1
        mem_smtr:               ds      1
        mem_tls:                ds      1
        mem_baca:               ds      1
        data_i2c:               ds      1

        flag_data_tls           equ     010h
        flag_data_baca          equ     011h
        awal_kiri               equ     012h
        ulang_kiri              equ     013h
        awal_kanan              equ     014h
        ulang_kanan             equ     015h
        tanda_memori            equ     016h

        org     10h
        cseg
pesan1:         db ' * menu utama *',0fh,'  pilihlah mode',0ffh
pesan2:         db ' * mode satu *',0fh,'  manual mode ',0ffh
pesan3:         db      '  * mode dua *',0fh,' automatic mode',0ffh
pesan4:         db      '  * mode tiga *',0fh,'  learning mode',0ffh

sda                     bit     p1.0
scl                     bit     p1.1
rs                      bit     p2.0
```

```
clk_lcd             bit     p2.2
phase_kiri          bit     p0.0
out_pwm1            bit     p0.1
phase_kanan         bit     p0.2
out_pwm2            bit     p0.3

;*****************************************
;       interupt
;=========================================
 int_kiri:
        clr     tr0
        clr     out_pwm1
        clr     ulang_kiri
        reti

int_kanan:
        clr     tr1
        clr     out_pwm2
        clr     ulang_kanan
        reti

;*****************************************
;rutin inisialisasi lcd
;input: clk_lcd=rs;
;bit 4~7 code
;bit 0 ~ rs, bit 2 ~ enable
;=========================================
inisial_lcd:
        mov     lcd_brs1,#80h
        mov     lcd_brs2,#0c0h
        acall time15
        mov     p2,#030h
        acall   clock_rst_lcd
        mov     p2,#030h
        acall   clock_rst_lcd
        mov     p2,#030h
        acall   clock_rst_lcd
        mov     p2,#020h
        acall   clock_rst_lcd
        mov     p2,#20h
        acall   clock_rst_lcd
        mov     p2,#80h
        acall   clock_rst_lcd
        mov     p2,#000h
        acall   clock_rst_lcd
        mov     p2,#60h
        acall   clock_rst_lcd
        mov     p2,#000h
        acall   clock_rst_lcd
```

```
        mov     p2,#0c0h
        acall   clock_rst_lcd
hapus_lcd:
        acall   time15
        mov     p2,#000h
        acall   clock_rst_lcd
        mov     p2,#10h
        acall   clock_rst_lcd
        ret

clock_rst_lcd:
        setb    clk_lcd
        clr     clk_lcd
        acall   time15
        ret

time15:
        push    00h
        push    01h
        mov     r0,#0ffh        ;time 15ms
        mov     r1,#01eh
time:   djnz    r0,time
        djnz    r1,time
        pop     01h
        pop     00h
        ret

;---------------------------------------------------------------
;rutin tampilkan karakter ke ke lcd
;r4 => reg. alamat kolom 1 pada lcd
;r4 (msb)~ alamat kolom 1
;r4 (lsb)~ don't care
;---------------------------------------------------------------
tulis_lcd:
        push    acc
        push    04h
        push    02h
        mov     r4,posisi_psn
        acall almt_awal
        acall kirim_psn
        pop     02h
        pop     04h
        pop     acc
        ret


;---------------------------------------------------------------
;sub-rutin untuk meletakkan kursor ke kolom 1
```

```
almt_awal:
        mov     p2,r4
        setb    clk_lcd
        clr     clk_lcd
        nop
        nop
        mov     p2,#000h
        setb    clk_lcd
        clr     clk_lcd
        acall   time15
        ret
;-------------------------------------------------
kirim_psn:
        mov     a,#00h
        movc    a,@a+dptr
        cjne    a,#0fh,kirim_lcd
lanjut_kirim:
        mov     r4,#0c0h
        acall   almt_awal
        inc     dptr
        sjmp    kirim_psn
;-------------------------------------------------
;sub-rutin kirim data ascii karakter ke lcd
kirim_lcd:
        cjne    a,#0ffh,lanjut
        sjmp    end_kirim
;-------------------------------------------------
;kirim data ascii karakter (msb) ke port2 (lsb)
lanjut:
        mov     r2,a
        anl     a,#0f0h
        add     a,#001h
        acall   delay_kirim
        mov     p2,a
        setb    clk_lcd
        clr     clk_lcd
        mov     a,r2
        swap    a
        anl     a,#0f0h
        add     a,#001h
        acall   delay_kirim
        mov     p2,a
        setb    clk_lcd
        clr     clk_lcd
        acall   delay_kirim
        inc     dptr
        sjmp    kirim_psn
end_kirim:
        ret
```

```
;-------------------------------------------------
;delay untuk kirim kode ascii
delay_kirim:
        push    00h
        mov     r0,#007h
        djnz    r0,$
        pop     00h
        ret

;*****************************************************
;       rutin   baca tulis eeprom i2c
;=====================================================
start_i2c:
        setb    sda
        setb    scl
        jnb     sda,busy
        jnb     scl,busy
        nop
        clr     sda
        nop
        nop
        nop
        nop
        nop
        clr     scl
        ret
busy:
        ret

;-------------------------------------------------
 kirim_i2c:
        push    acc
        push    b
        mov     a,data_i2c
        mov     b,#8
ulang_kirim:
        rlc     a
        mov     sda,c
        nop
        setb    scl
        nop
        nop
        nop
        nop
        clr     scl
        djnz    b,ulang_kirim
        pop     b
        pop     acc
        setb    sda
```

```
        nop
        nop
        setb    scl
        nop
        nop
        nop
        nop
        mov     c,sda
        jc      keluar
        clr     scl
        ret
keluar:
        ret

;-----------------------------------------------
terima_i2c:
        push    acc
        push    b
        mov     data_i2c,#0
        setb    sda
        mov     b,#8
ulang1:
        nop
        nop
        nop
        setb    scl
        nop
        nop
        mov     c,sda
        rlc     a
        clr     scl
        djnz    b,ulang1
        mov     data_i2c,a
        pop     b
        pop     acc
        ret

stop_i2c:
        clr     sda
        nop
        nop
        setb    scl
        nop
        nop
        nop
        nop
        setb    sda
        ret
```

```
;-----------------------------------------------
delay_lama:
        push    07h
        push    06h
        push    05h
        mov     r7,#001h
delay1:
        mov     r6,#00fh
delay2:
        mov     r5,#0ffh
        djnz    r5,$
        djnz    r6,delay2
        djnz    r7,delay1
        pop     05h
        pop     06h
        pop     07h
        ret

;-----------------------------------------------
simpan_track:
        acall   start_i2c
        mov     data_i2c,#0a0h
        acall   kirim_i2c
        mov     dph,almt_mem_msb
        mov     dpl,almt_mem_lsb
        mov     data_i2c,dph
        acall   kirim_i2c
        mov     data_i2c,dpl
        acall   kirim_i2c
        mov     data_i2c,mem_tls
        acall   kirim_i2c
        acall   stop_i2c
        acall   time15
        inc     dptr
        mov     almt_mem_msb,dph
        mov     almt_mem_lsb,dpl
        ret

;-----------------------------------------------
reset:
        setb    sda
cari_reset:
        nop
        setb    scl
        jb      sda,end_reset
        clr     scl
        nop
        nop
        sjmp    cari_reset
```

```
end_reset:
        ret

;-------------------------------------------------
baca_track:
        acall   reset
        acall   start_i2c
        mov     data_i2c,#0a0h
        acall   kirim_i2c
        mov     dph,almt_mem_msb
        mov     dpl,almt_mem_lsb
        mov     data_i2c,dph
        acall   kirim_i2c
        mov     data_i2c,dpl
        acall   kirim_i2c
        acall   start_i2c
        mov     data_i2c,#0a1h
        acall   kirim_i2c
        acall   terima_i2c
        acall   stop_i2c
        mov     mem_baca,data_i2c
        inc     dptr
        mov     almt_mem_msb,dph
        mov     almt_mem_lsb,dpl
        ret

;-------------------------------------------------------------------------------
; rutin  pengkodean track ke memori
; data yang siap ditulis ke memori disimpan ke mem_tls
; flag_data_tls = clr bit ~       data siap u/ disimpan
;-------------------------------------------------------------------------------
kode_data_mem:
        push    acc
        jnb     flag_data_tls,alih1
        mov     a,#0
        mov     a,arah_kanan
        rl      a
        rl      a
        add     a,arah_kiri
        mov     mem_smtr,a
        clr     flag_data_tls
        sjmp    end_kode
alih1:
        mov     a,#0
        mov     a,arah_kanan
        rl      a
        rl      a
        add     a,arah_kiri
        swap    a
```

```
        add     a,mem_smtr
        mov     mem_tls,a
;data siap untuk dikirim ke memori
        setb    flag_data_tls
        acall   simpan_track
end_kode:
        pop     acc
        ret

;*****************************************************
; rutin baca input remote kontrol
; output dari remote dihubungkan pada port p1 msb
; p3.4~ ;p3.5~  ;p3.6~  ;p3.7~
;=====================================================
baca_remote:
        push    acc
        push    07h
        mov     a,p3
        anl     a,#030h
        swap    a
        cjne    a,#02,kiri_swap
        mov     arah_kiri,#01h
        mov     arah_kanan,#01h
        sjmp    detek_kanan
kiri_swap:
        cjne    a,#01h,kiri_swap1
        mov     arah_kiri,#02h
        mov     arah_kanan,#02h
        sjmp    detek_kanan
kiri_swap1:
        mov     arah_kiri,#00h
        mov     arah_kanan,#00h
        sjmp    end_read
detek_kanan:
        mov     a,p3
        anl     a,#0c0h
        swap    a
        mov     r7,a
        cjne    a,#008h,kanan_swap1
        mov     arah_kanan,#00h
        sjmp    end_read
kanan_swap1:
        mov     a,r7
        cjne    a,#004h,end_read
        mov     arah_kiri,#00h
end_read:
        pop     07h
        pop     acc
        ret
```

```
;-------------------------------------------------------------------------
; rutin kontrol driver motor dc
; sinyal kontrol terdapat pada port p0 lsb
; p0.0 ~ ('1') motor kiri ccw          ;p0.1 ~ ('1') irf kiri aktif
; p0.2 ~ ('1') motor kanan cw          ;p0.3 ~ ('1') irf kanan aktif
;=========================================================================
pwm:
        push    acc
        mov     a,p0
        anl     a,#0f0h
        mov     p0,a
        mov     a,arah_kiri
        cjne    a,#02,ktrl_kanan
        setb    phase_kiri
ktrl_kanan:
        mov     a,arah_kanan
        cjne    a,#01,end_control
        setb    phase_kanan
end_control:
        pop     acc
        ret
;-------------------------------------------------
;rutin pembangkit 2 outputan pwm
;-------------------------------------------------
pwm_kiri:
        clr     tr0
        jnb     ulang_kiri,end_pwm_kiri
        jnb     awal_kiri,pwm_kiri1
        mov     th0,#0f7h
        mov     tl0,#0cch
        clr     awal_kiri
        setb    tr0                  ;aktif siklus
        setb    out_pwm1
        ret
pwm_kiri1:
        mov     th0,#0ech
        mov     tl0,#0dch
        setb    awal_kiri
        setb    tr0                  ;off siklus
        clr     out_pwm1
        ret
end_pwm_kiri:
        clr     out_pwm1
        clr     phase_kiri
        ret
pwm_kanan:
        clr     tr1
        jnb     ulang_kanan,end_pwm_kanan
```

```
        jnb     awal_kanan,pwm_kanan1
        clr     tr1
        mov     th1,#0f6h
        mov     tl1,#0fah
        clr     awal_kanan
        setb    tr1                     ;aktif siklus
        setb    out_pwm2
        ret
pwm_kanan1:
        mov     th1,#0edh
        mov     tl1,#0aeh
        setb    awal_kanan
        setb    tr1                     ;off    siklus
        clr     out_pwm2
        ret
end_pwm_kanan:
        clr     out_pwm2
        clr     phase_kanan
        ret

;-------------------------------------------------
deteksi_pwm:
        push    acc
        mov     tmod,#11h
        mov     a,arah_kiri
        jz      aktif_kanan
        setb    ulang_kiri
        setb    awal_kiri
        acall   pwm_kiri
aktif_kanan:
        mov     a,arah_kanan
        jz      end_deteksi_pwm
        setb    ulang_kanan
        setb    awal_kanan
        acall   pwm_kanan
end_deteksi_pwm:
        pop     acc
        ret
;-------------------------------------------------
sinyal_pwm_man:
        acall   deteksi_pwm
masih_man:
        jb      ulang_kiri,masih_man
        jb      ulang_kanan,masih_man
        acall   delay_lama
        ret

;-------------------------------------------------
sinyal_pwm_oto:
```

```
        acall   deteksi_pwm
        jb      tanda_memori,masih_oto
        acall   baca_track
masih_oto:
        jb      ulang_kiri,masih_oto
        jb      ulang_kanan,masih_oto
        acall   delay_lama
        ret

;-------------------------------------------------
sinyal_pwm_learn:
        acall   deteksi_pwm
        acall   kode_data_mem
masih_learn:
        jb      ulang_kiri,masih_learn
        jb      ulang_kanan,masih_learn
        acall   delay_lama
        acall   delay_lama
        ret

;-------------------------------------------------
;       mode program manual
;=================================================
mode_satu:
        push    acc
ulang_mode_1:
        acall   baca_remote
        mov     a,arah_kiri
        swap    a
        add     a,arah_kanan
        jz      end_mode_satu
        acall   pwm
        acall   sinyal_pwm_man
end_mode_satu:
        jb      p1.5,ulang_mode_1
        pop     acc
        ret

;-------------------------------------------------
;       mode program otomatis
;=================================================
mode_dua:
        push    acc
        mov     almt_mem_msb,#00h
        mov     almt_mem_lsb,#00h
        acall   baca_track
ulang_mode2:
        setb    tanda_memori
        mov     a,mem_baca
```

```
        anl     a,#00fh
        jz      end_mode_dua
        mov     arah_kiri,a
        anl     arah_kiri,#03h
        anl     a,#00ch
        rr      a
        rr      a
        mov     arah_kanan,a
sa1:
        jnb     p3.0,sa11
        sjmp    end_sa1
sa11:
        mov     dptr,#pesan5
        acall   psn_halangan
        acall   halangan
end_sa1:
        acall   pwm
        acall   sinyal_pwm_oto
;jalankan memori track msb
        clr     tanda_memori
        mov     a,mem_baca
        swap    a
        anl     a,#00fh
        jz      end_mode_dua
        mov     arah_kiri,a
        anl     arah_kiri,#03h
        anl     a,#00ch
        rr      a
        rr      a
        mov     arah_kanan,a
sa2:
        jnb     p3.0,sa21
        sjmp    end_sa2
sa21:
        mov     dptr,#pesan5
        acall   psn_halangan
        acall   halangan
end_sa2:
        acall   pwm
        acall   sinyal_pwm_oto
        jnb     p1.5,end_mode_dua
        sjmp    ulang_mode2
end_mode_dua:
        pop     acc
        ret

psn_halangan:
        acall   hapus_lcd
        mov     posisi_psn,lcd_brs1
```

```
        acall   tulis_lcd
        ret

halangan:
        jnb     p3.0,$
        acall   delay_lama
        acall   delay_lama
        jnb     p3.0,halangan
        mov     dptr,#pesan3
        acall   psn_halangan
        ret
;-----------------------------------------------
; mode program learning
;===============================================
mode_tiga:
        push    acc
        mov     almt_mem_msb,#0
        mov     almt_mem_lsb,#0
;inisialisasi   flag memori tulis
        setb    flag_data_tls
ulang_mode3:
        acall   baca_remote
        mov     a,arah_kiri
        swap    a
        add     a,arah_kanan
        jz      end_mode_tiga
        acall   pwm
        acall   sinyal_pwm_learn
end_mode_tiga:
        jb      p1.5,ulang_mode3
        acall   delay_lama
        jb      flag_data_tls,akhir_tls1
        mov     mem_tls,mem_smtr
        acall   simpan_track
        sjmp    akhir_tls2
akhir_tls1:
        mov     mem_tls,#00h
        acall   simpan_track
akhir_tls2:
        pop     acc
        ret

;-----------------------------------------------
;       rutin inisialisasi yang diperlukan
;===============================================
inisialisasi:
        mov     p1,#0ffh
        mov     p0,#00h
        mov     ie,#8fh
```

```
        setb    it0
        setb    it1
        acall inisial_lcd
        ret


;**********************************
;       main   program
;==================================
start:
        mov     sp,#2ah
        acall   inisialisasi
prog_utama:
        acall   hapus_lcd
        mov     posisi_psn,lcd_brs1
        mov     dptr,#pesan1
        acall   tulis_lcd
baca_keypad:
        jb      p1.2,cek_keypad1
        acall   hapus_lcd
        mov     posisi_psn,lcd_brs1
        mov     dptr,#pesan2
        acall   tulis_lcd
        acall   mode_satu
        sjmp    prog_utama
cek_keypad1:
        jb      p1.3,cek_keypad2
        acall   hapus_lcd
        mov     posisi_psn,lcd_brs1
        mov     dptr,#pesan3
        acall   tulis_lcd
        acall   mode_dua
        sjmp    prog_utama
cek_keypad2:
        jb      p1.4,baca_keypad
        acall   hapus_lcd
        mov     posisi_psn,lcd_brs1
        mov     dptr,#pesan4
        acall   tulis_lcd
        acall   mode_tiga
        sjmp    prog_utama
        end
```

Features

- Compatible with MCS®-51 Products
- 4K Bytes of In-System Programmable (ISP) Flash Memory
  - Endurance: 1000 Write/Erase Cycles
- 4.0V to 5.5V Operating Range
- Fully Static Operation: 0 Hz to 33 MHz
- Three-level Program Memory Lock
- 128 x 8-bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-bit Timer/Counters
- Six Interrupt Sources
- Full Duplex UART Serial Channel
- Low-power Idle and Power-down Modes
- Interrupt Recovery from Power-down Mode
- Watchdog Timer
- Dual Data Pointer
- Power-off Flag
- Fast Programming Time
- Flexible ISP Programming (Byte and Page Mode)

Description

The AT89S51 is a low-power, high-performance CMOS 8-bit microcontroller with 4K bytes of In-System Programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT89S51 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

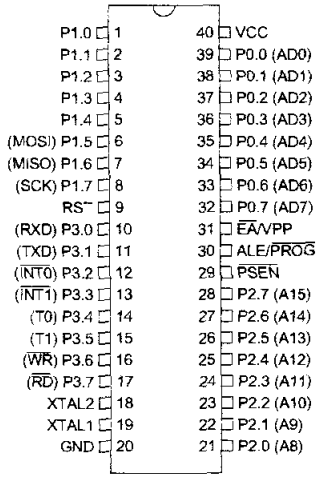
8-bit Microcontroller with 4K Bytes In-System Programmable Flash

AT89S51

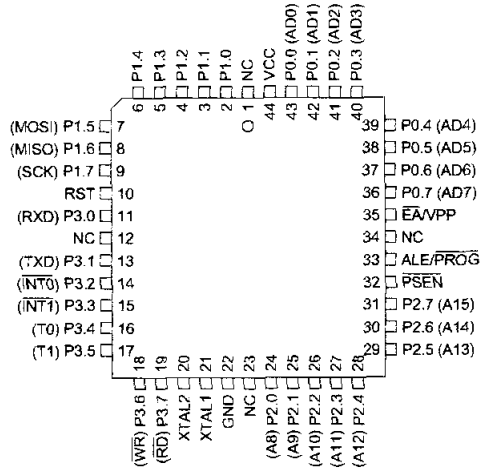
2487B–MICRO–12/03

Pin Configurations

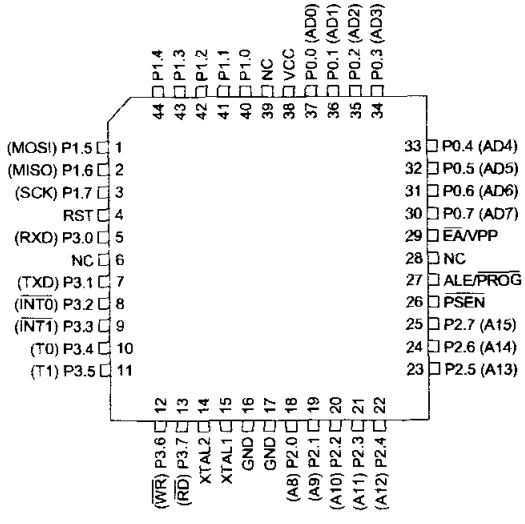
PDIP



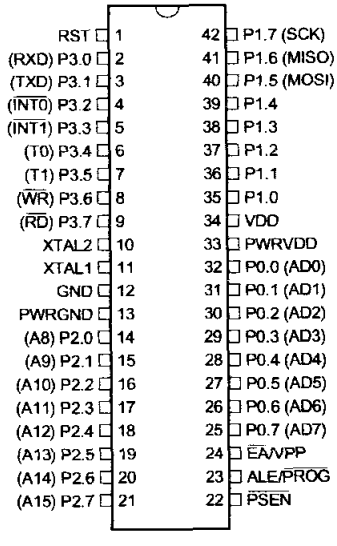
PLCC



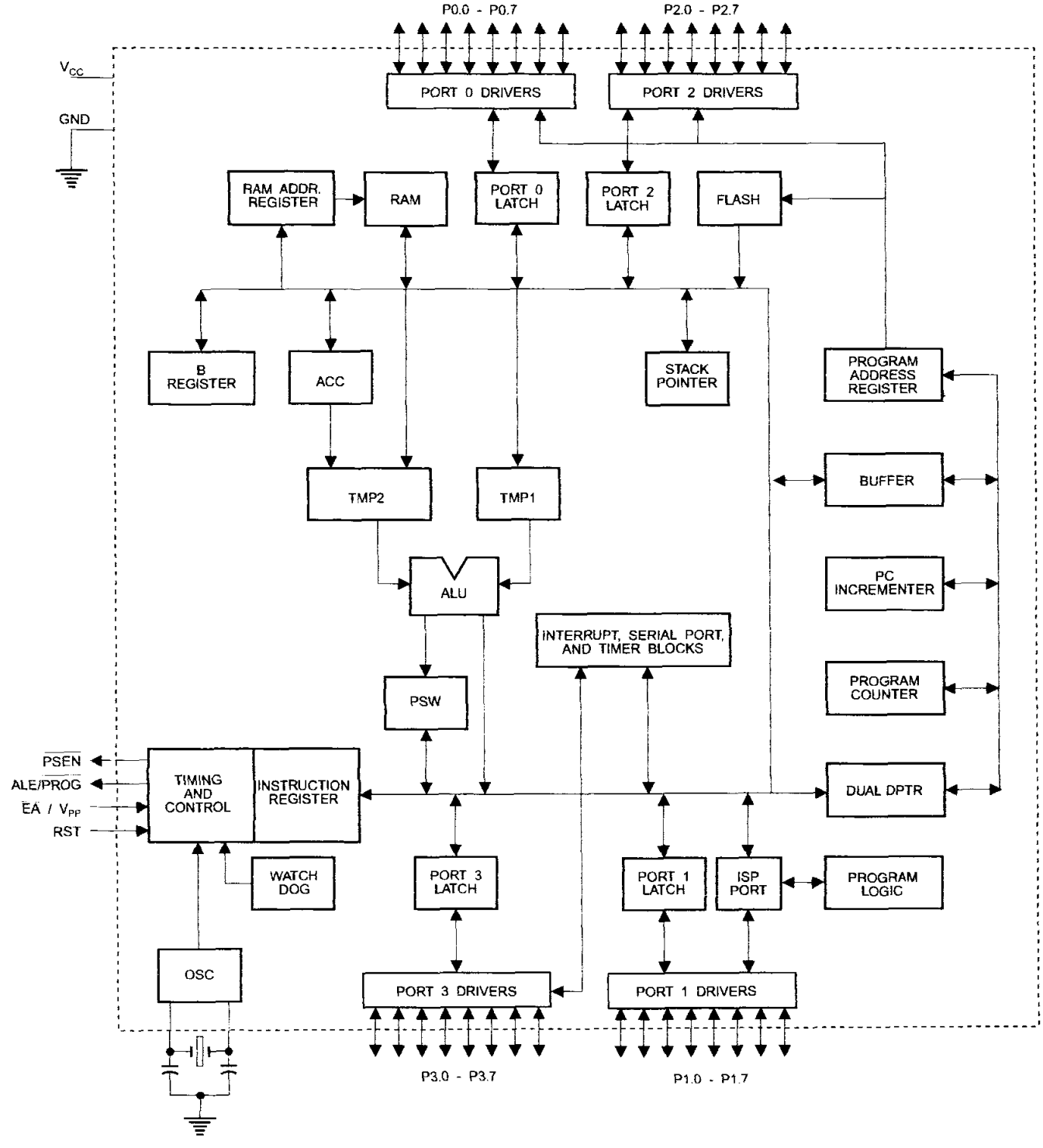
TQFP



PDIP



Block Diagram



Pin Description

VCC Supply voltage (all packages except 42-PDIP).

GND Ground (all packages except 42-PDIP; for 42-PDIP GND connects only the logic core and the embedded program memory).

VDD Supply voltage for the 42-PDIP which connects only the logic core and the embedded program memory.

PWRVDD Supply voltage for the 42-PDIP which connects only the I/O Pad Drivers. The application board **MUST** connect both VDD and PWRVDD to the board supply voltage.

PWRGND Ground for the 42-PDIP which connects only the I/O Pad Drivers. PWRGND and GND are weakly connected through the common silicon substrate, but not through any metal link. The application board **MUST** connect both GND and PWRGND to the board ground.

Port 0 Port 0 is an 8-bit open drain bi-directional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pull-ups.

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. **External pull-ups are required during program verification.**

Port 1 Port 1 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}) because of the internal pull-ups.

Port 1 also receives the low-order address bytes during Flash programming and verification.

Port Pin	Alternate Functions
P1.5	MOSI (used for In-System Programming)
P1.6	MISO (used for In-System Programming)
P1.7	SCK (used for In-System Programming)

Port 2 Port 2 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pull-ups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3

Port 3 is an 8-bit bi-directional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pull-ups.

Port 3 receives some control signals for Flash programming and verification.

Port 3 also serves the functions of various special features of the AT89S51, as shown in the following table.

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives High for 98 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.

ALE/$\overline{PROG}$

Address Latch Enable (ALE) is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{PROG}$) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

$\overline{PSEN}$

Program Store Enable ($\overline{PSEN}$) is the read strobe to external program memory.

When the AT89S51 is executing code from external program memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external data memory.

$\overline{EA}$/VPP

External Access Enable. $\overline{EA}$ must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, $\overline{EA}$ will be internally latched on reset.

$\overline{EA}$ should be strapped to V_{CC} for internal program executions.

This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier

Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 1.

Note that not all of the addresses are occupied, and unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

Table 1. AT89S51 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000								0D7H
0C8H									0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000								0AFH
0A0H	P2 11111111		AUXR1 XXXXXXX0				WDTRST XXXXXXXX		0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111								97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR XXX00XX0		8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000		PCON 0XXX0000	87H
	0	1	2	3	4	5	6	7	

Features

- Low Voltage and Standard Voltage Operation
  - 5.0 (V_{CC} = 4.5V to 5.5V)
  - 2.7 (V_{CC} = 2.7V to 5.5V)
  - 1.8 (V_{CC} = 1.8V to 3.6V)
- Internally Organized 16,384 x 8 and 32,768 x 8
- 2-Wire Serial Interface
- Schmitt Trigger, Filtered Inputs for Noise Suppression
- Bidirectional Data Transfer Protocol
- 1 MHz (5V), 400 kHz (2.7V) and 100 kHz (1.8V) Compatibility
- Write Protect Pin for Hardware and Software Data Protection
- 64-Byte Page Write Mode (Partial Page Writes Allowed)
- Self-Timed Write Cycle (5 ms typical)
- High Reliability
  - Endurance: 100,000 Write Cycles
  - Data Retention: 40 Years
  - ESD Protection: > 4000V
- Automotive Grade and Extended Temperature Devices Available
- 8-Pin JEDEC PDIP, 8-Pin JEDEC and EIAJ SOIC, 14-Pin TSSOP, and 8-Pin Leadless Array Packages

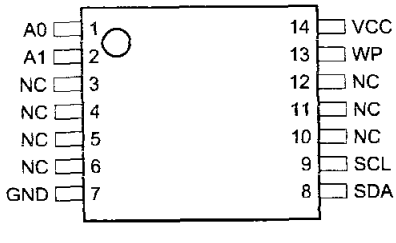
Description

The AT24C128/256 provides 131,072/262,144 bits of serial electrically erasable and programmable read only memory (EEPROM) organized as 16,384/32,768 words of 8 bits each. The device's cascadable feature allows up to 4 devices to share a common 2-wire bus. The device is optimized for use in many industrial and commercial applications where low power and low voltage operation are essential. The devices are available in space-saving 8-pin JEDEC PDIP, 8-pin EIAJ, 8-pin JEDEC SOIC, 14-pin TSSOP, and 8-pin LAP packages. In addition, the entire family is available in 5.0V (4.5V to 5.5V), 2.7V (2.7V to 5.5V) and 1.8V (1.8V to 3.6V) versions.

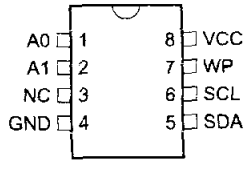
Pin Configurations

Pin Name	Function
A_0 to A_1	Address Inputs
SDA	Serial Data
SCL	Serial Clock Input
WP	Write Protect
NC	No Connect

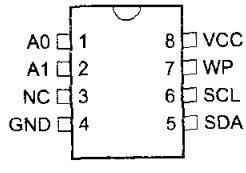
14-Pin TSSOP



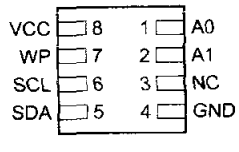
8-Pin PDIP



8-Pin SOIC



8-Pin Leadless Array



Bottom View

2-Wire Serial EEPROMs

128K (16,384 x 8)

256K (32,768 x 8)

AT24C128
AT24C256

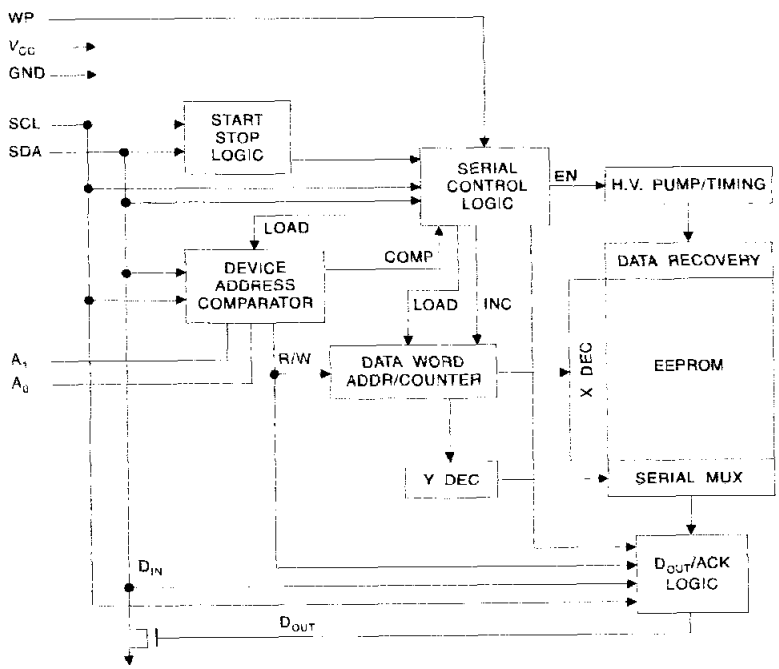
Rev. 0670C–08/98

Absolute Maximum Ratings*

Operating Temperature	-55°C to +125°C
Storage Temperature	-65°C to +150°C
Voltage on Any Pin with Respect to Ground	-1.0V to +7.0V
Maximum Operating Voltage	6.25V
DC Output Current	5.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Block Diagram



Pin Description

SERIAL CLOCK (SCL): The SCL input is used to positive edge clock data into each EEPROM device and negative edge clock data out of each device.

SERIAL DATA (SDA): The SDA pin is bidirectional for serial data transfer. This pin is open-drain driven and may be wire-ORed with any number of other open-drain or open collector devices.

DEVICE/PAGE ADDRESSES (A1, A0): The A1 and A0 pins are device address inputs that are hardwired or left not connected for hardware compatibility with AT24C32/64. When the pins are hardwired, as many as four 128K/256K devices may be addressed on a single bus system (device addressing is discussed in detail under the Device Addressing section). When the pins are not hardwired, the default A_1 and A_0 are zero.

WRITE PROTECT (WP): The write protect input, when tied to GND, allows normal write operations. When WP is tied high to V_{CC}, all write operations to the memory are inhibited. If left unconnected, WP is internally pulled down to GND. Switching WP to V_{CC} prior to a write operation creates a software write protect function.

Memory Organization

AT24C128/256, 128K/256K SERIAL EEPROM: The 128K/256K is internally organized as 256/512 pages of 64-bytes each. Random word addressing requires a 14/15-bit data word address.

Pin Capacitance[1]

Applicable over recommended operating range from T_A = 25°C, f = 1.0 MHz, V_{CC} = +1.8V.

Symbol	Test Condition	Max	Units	Conditions
$C_{I/O}$	Input/Output Capacitance (SDA)	8	pF	$V_{I/O}$ = 0V
C_{IN}	Input Capacitance (A_0, A_1, SCL)	6	pF	V_{IN} = 0V

Note: This parameter is characterized and is not 100% tested.

DC Characteristics

Applicable over recommended operating range from: T_{AI} = -40°C to +85°C, V_{CC} = +1.8V to +5.5V, T_{AC} = 0°C to +70°C, V_{CC} = +1.8V to +5.5V (unless otherwise noted).

Symbol	Parameter		Test Condition	Min	Typ	Max	Units
V_{CC1}	Supply Voltage			1.8		3.6	V
V_{CC2}	Supply Voltage			2.7		5.5	V
V_{CC3}	Supply Voltage			4.5		5.5	V
I_{CC1}	Supply Current	V_{CC} = 5.0V	READ at 400 kHz		1.0	2.0	mA
I_{CC2}	Supply Current	V_{CC} = 5.0V	WRITE at 400 kHz		2.0	3.0	mA
I_{SB1}	Standby Current (1.8V option)	V_{CC} = 1.8V	V_{IN} = V_{CC} or V_{SS}			0.2	µA
		V_{CC} = 3.6V				2.0	
I_{SB2}	Standby Current (2.7V option)	V_{CC} = 2.7V	V_{IN} = V_{CC} or V_{SS}			0.5	µA
		V_{CC} = 5.5V				6.0	
I_{SB3}	Standby Current (5.0V option)	V_{CC} = 4.5 - 5.5V	V_{IN} = V_{CC} or V_{SS}			6.0	µA
I_{LI}	Input Leakage Current		V_{IN} = V_{CC} or V_{SS}		0.10	3.0	µA
I_{LO}	Output Leakage Current		V_{OUT} = V_{CC} or V_{SS}		0.05	3.0	µA
V_{IL}	Input Low Level[Note:]			-0.6		V_{CC} x 0.3	V
V_{IH}	Input High Level[Note:]			V_{CC} x 0.7		V_{CC} + 0.5	V
V_{OL2}	Output Low Level	V_{CC} = 3.0V	I_{OL} = 2.1 mA			0.4	V
V_{OL1}	Output Low Level	V_{CC} = 1.8V	I_{OL} = 0.15 mA			0.2	V

Note: V_{IL} min and V_{IH} max are reference only and are not tested

AC Characteristics

Applicable over recommended operating range from T_A = -40°C to +85°C, V_{CC} = +1.8V to +5.5V, CL = 100 pF (unless otherwise noted). Test conditions are listed in Note 2.

Symbol	Parameter	1.8-volt		2.7-volt		5.0-volt		Units
		Min	Max	Min	Max	Min	Max	
f_{SCL}	Clock Frequency, SCL		100		400		1000	kHz
t_{LOW}	Clock Pulse Width Low	4.7		1.3		0.6		µs
t_{HIGH}	Clock Pulse Width High	4.0		1.0		0.4		µs
t_{AA}	Clock Low to Data Out Valid	0.1	4.5	0.05	0.9	0.05	0.55	µs
t_{BUF}	Time the bus must be free before a new transmission can start(1)	4.7		1.3		0.5		µs
$t_{HD.STA}$	Start Hold Time	4.0		0.6		0.25		µs
$t_{SU.STA}$	Start Set-up Time	4.7		0.6		0.25		µs
$t_{HD.DAT}$	Data In Hold Time	0		0		0		µs
$t_{SU.DAT}$	Data In Set-up Time	200		100		100		ns
t_R	Inputs Rise Time(1)		1.0		0.3		0.3	µs
t_F	Inputs Fall Time(1)		300		300		100	ns
$t_{SU.STO}$	Stop Set-up Time	4.7		0.6		0.25		µs
t_{DH}	Data Out Hold Time	100		50		50		ns
t_{WR}	Write Cycle Time		20		10		10	ms
Endurance(1)	5.0V, 25°C, Page Mode	100K		100K		100K		Write Cycles

Notes:
1. This parameter is characterized and is not 100% tested.
2. AC measurement conditions:
   R_L (connects to V_{CC}): 1.3KΩ (2.7V, 5V), 10KΩ (1.8V)
   Input pulse voltages: $0.3V_{CC}$ to $0.7V_{CC}$
   Input rise and fall times: ≤50ns
   Input and output timing reference voltages: $0.5V_{CC}$

Device Operation

CLOCK and DATA TRANSITIONS: The SDA pin is normally pulled high with an external device. Data on the SDA pin may change only during SCL low time periods (refer to Data Validity timing diagram). Data changes during SCL high periods will indicate a start or stop condition as defined below.

START CONDITION: A high-to-low transition of SDA with SCL high is a start condition which must precede any other command (refer to Start and Stop Definition timing diagram).

STOP CONDITION: A low-to-high transition of SDA with SCL high is a stop condition. After a read sequence, the stop command will place the EEPROM in a standby power mode (refer to Start and Stop Definition timing diagram).

ACKNOWLEDGE: All addresses and data words are serially transmitted to and from the EEPROM in 8-bit words. The EEPROM sends a zero during the ninth clock cycle to acknowledge that it has received each word.

STANDBY MODE: The AT24C128/256 features a low power standby mode which is enabled: a) upon power-up and b) after the receipt of the STOP bit and the completion of any internal operations.

MEMORY RESET: After an interruption in protocol, power loss or system reset, any 2-wire part can be reset by following these steps: (a) Clock up to 9 cycles, (b) look for SDA high in each cycle while SCL is high and then (c) create a start condition as SDA is high.

National Semiconductor

February 1995

LM555/LM555C Timer

General Description

The LM555 is a highly stable device for generating accurate time delays or oscillation. Additional terminals are provided for triggering or resetting if desired. In the time delay mode of operation, the time is precisely controlled by one external resistor and capacitor. For astable operation as an oscillator, the free running frequency and duty cycle are accurately controlled with two external resistors and one capacitor. The circuit may be triggered and reset on falling waveforms, and the output circuit can source or sink up to 200 mA or drive TTL circuits.

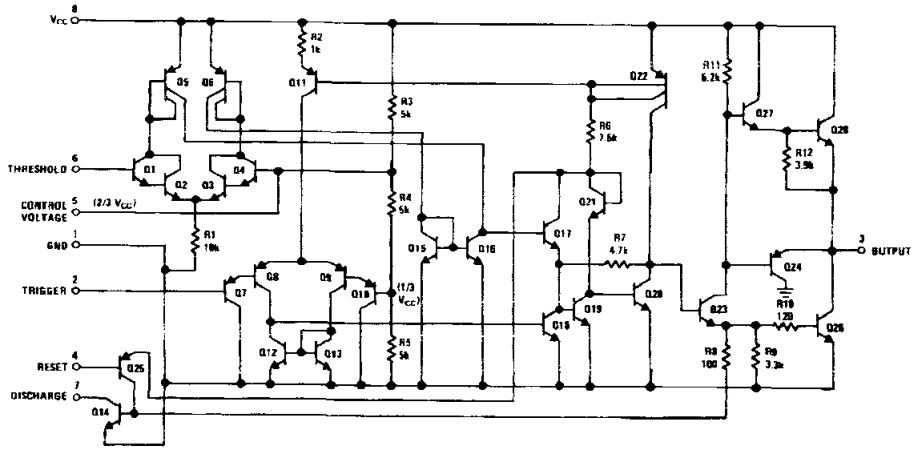
Features

- Direct replacement for SE555/NE555
- Timing from microseconds through hours
- Operates in both astable and monostable modes
- Adjustable duty cycle
- Output can source or sink 200 mA
- Output and supply TTL compatible
- Temperature stability better than 0.005% per °C
- Normally on and normally off output

Applications

- Precision timing
- Pulse generation
- Sequential timing
- Time delay generation
- Pulse width modulation
- Pulse position modulation
- Linear ramp generator

Schematic Diagram



TL/H/7851–1

RRD-B30M115/Printed in U. S. A.

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage +18V

Power Dissipation (Note 1)
- LM555H, LM555CH 760 mW
- LM555, LM555CN 1180 mW

Operating Temperature Ranges
- LM555C 0°C to +70°C
- LM555 −55°C to +125°C

Storage Temperature Range −65°C to +150°C

Soldering Information
- Dual-In-Line Package
  - Soldering (10 Seconds) 260°C
- Small Outline Package
  - Vapor Phase (60 Seconds) 215°C
  - Infrared (15 Seconds) 220°C

See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.

Electrical Characteristics (T_A = 25°C, V_{CC} = +5V to +15V, unless otherwise specified)

Parameter	Conditions	Limits						Units
		LM555			LM555C			
		Min	Typ	Max	Min	Typ	Max	
Supply Voltage		4.5		18	4.5		16	V
Supply Current	V_{CC} = 5V, R_L = ∞ V_{CC} = 15V, R_L = ∞ (Low State) (Note 2)		3 10	5 12		3 10	6 15	mA mA
Timing Error, Monostable Initial Accuracy Drift with Temperature Accuracy over Temperature Drift with Supply	 R_A = 1k to 100 kΩ, C = 0.1 μF, (Note 3)		 0.5 30 1.5 0.05			 1 50 1.5 0.1		 % ppm/°C % %/V
Timing Error, Astable Initial Accuracy Drift with Temperature Accuracy over Temperature Drift with Supply	 R_A, R_B = 1k to 100 kΩ, C = 0.1 μF, (Note 3)		 1.5 90 2.5 0.15			 2.25 150 3.0 0.30		 % ppm/°C % %/V
Threshold Voltage			0.667			0.667		x V_{CC}
Trigger Voltage	V_{CC} = 15V V_{CC} = 5V	4.8 1.45	5 1.67	5.2 1.9		5 1.67		V V
Trigger Current			0.01	0.5		0.5	0.9	μA
Reset Voltage		0.4	0.5	1	0.4	0.5	1	V
Reset Current			0.1	0.4		0.1	0.4	mA
Threshold Current	(Note 4)		0.1	0.25		0.1	0.25	μA
Control Voltage Level	V_{CC} = 15V V_{CC} = 5V	9.6 2.9	10 3.33	10.4 3.8	9 2.6	10 3.33	11 4	V V
Pin 7 Leakage Output High			1	100		1	100	nA
Pin 7 Sat (Note 5) Output Low Output Low	 V_{CC} = 15V, I_7 = 15 mA V_{CC} = 4.5V, I_7 = 4.5 mA		 150 70	 100		 180 80	 200	 mV mV

Electrical Characteristics $T_A = 25°C$, $V_{CC} = +5V$ to $+15V$, (unless othewise specified) (Continued)

Parameter	Conditions	Limits						Units
		LM555			LM555C			
		Min	Typ	Max	Min	Typ	Max	
Output Voltage Drop (Low)	$V_{CC} = 15V$							
	$I_{SINK} = 10$ mA		0.1	0.15		0.1	0.25	V
	$I_{SINK} = 50$ mA		0.4	0.5		0.4	0.75	V
	$I_{SINK} = 100$ mA		2	2.2		2	2.5	V
	$I_{SINK} = 200$ mA		2.5			2.5		V
	$V_{CC} = 5V$							
	$I_{SINK} = 8$ mA		0.1	0.25				V
	$I_{SINK} = 5$ mA					0.25	0.35	V
Output Voltage Drop (High)	$I_{SOURCE} = 200$ mA, $V_{CC} = 15V$		12.5			12.5		V
	$I_{SOURCE} = 100$ mA, $V_{CC} = 15V$	13	13.3		12.75	13.3		V
	$V_{CC} = 5V$	3	3.3		2.75	3.3		V
Rise Time of Output			100			100		ns
Fall Time of Output			100			100		ns

Note 1: For operating at elevated temperatures the device must be derated above 25°C based on a +150°C maximum junction temperature and a thermal resistance of 164°c/w (TO-5), 106°c/w (DIP) and 170°c/w (SO-8) junction to ambient.

Note 2: Supply current when output high typically 1 mA less at $V_{CC} = 5V$.

Note 3: Tested at $V_{CC} = 5V$ and $V_{CC} = 15V$.

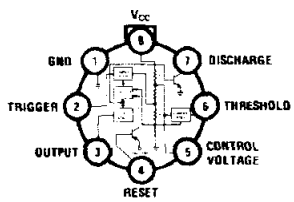
Note 4: This will determine the maximum value of $R_A + R_B$ for 15V operation. The maximum total ($R_A + R_B$) is 20 MΩ.

Note 5: No protection against excessive pin 7 current is necessary providing the package dissipation rating will not be exceeded.

Note 6: Refer to RETS555X drawing of military LM555H and LM555J versions for specifications.

Connection Diagrams

Metal Can Package

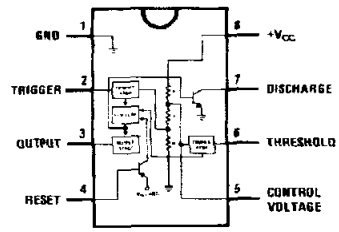


TL/H/7851–2

Top View

Order Number LM555H or LM555CH
See NS Package Number H08C

Dual-In-Line and Small Outline Packages



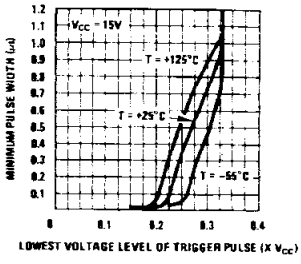
TL/H/7851–3

Top View

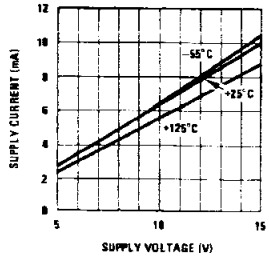
Order Number LM555J, LM555CJ, LM555CM or LM555CN
See NS Package Number J08A, M08A or N08E

Typical Performance Characteristics

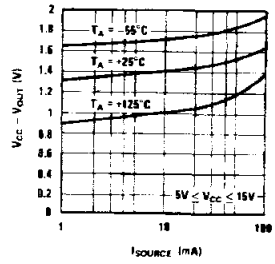
Minimuim Pulse Width Required for Triggering



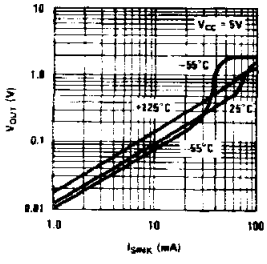
Supply Current vs Supply Voltage



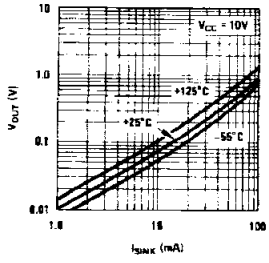
High Output Voltage vs Output Source Current



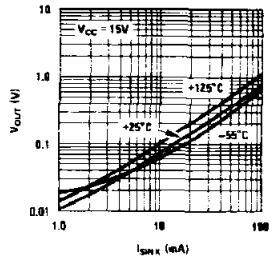
Low Output Voltage vs Output Sink Current



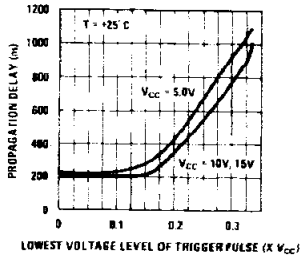
Low Output Voltage vs Output Sink Current



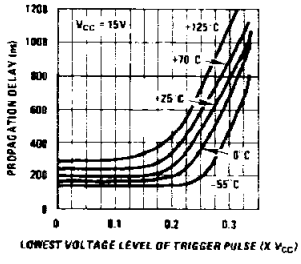
Low Output Voltage vs Output Sink Current



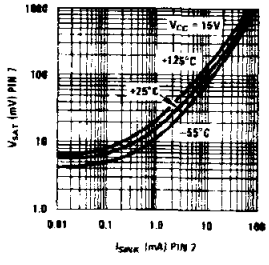
Output Propagation Delay vs Voltage Level of Trigger Pulse



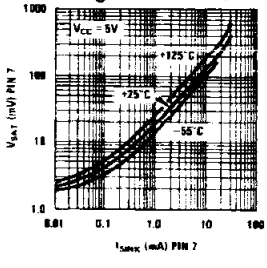
Output Propagation Delay vs Voltage Level of Trigger Pulse



Discharge Transistor (Pin 7) Voltage vs Sink Current



Discharge Transistor (Pin 7) Voltage vs Sink Current



TL/H/7851–4

FAIRCHILD SEMICONDUCTOR®

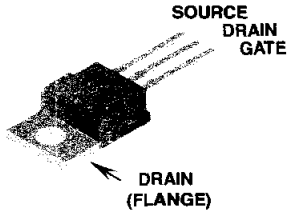
IRF540N

Data Sheet | January 2002

33A, 100V, 0.040 Ohm, N-Channel, Power MOSFET

Packaging

JEDEC TO-220AB

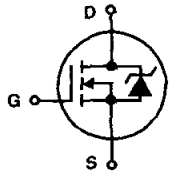


IRF540N

Features

- Ultra Low On-Resistance
  - $r_{DS(ON)} = 0.040\Omega$, $V_{GS} = 10V$
- Simulation Models
  - Temperature Compensated PSPICE™ and SABER© Electrical Models
  - Spice and SABER© Thermal Impedance Models
  - www.fairchildsemi.com
- Peak Current vs Pulse Width Curve
- UIS Rating Curve

Symbol



Ordering Information

PART NUMBER	PACKAGE	BRAND
IRF540N	TO-220AB	IRF540N

Absolute Maximum Ratings $T_C = 25^oC$, Unless Otherwise Specified

		IRF540N	UNITS
Drain to Source Voltage (Note 1)	V_{DSS}	100	V
Drain to Gate Voltage ($R_{GS} = 20k\Omega$) (Note 1)	V_{DGR}	100	V
Gate to Source Voltage	V_{GS}	±20	V
Drain Current			
Continuous ($T_C = 25^oC$, $V_{GS} = 10V$) (Figure 2)	I_D	33	A
Continuous ($T_C = 100^oC$, $V_{GS} = 10V$) (Figure 2)	I_D	23	A
Pulsed Drain Current	I_{DM}	Figure 4	
Pulsed Avalanche Rating	UIS	Figures 6, 14, 15	
Power Dissipation	P_D	120	W
Derate Above 25^oC		0.80	$W/^oC$
Operating and Storage Temperature	T_J, T_{STG}	-55 to 175	^oC
Maximum Temperature for Soldering			
Leads at 0.063in (1.6mm) from Case for 10s	T_L	300	^oC
Package Body for 10s, See Techbrief TB334	T_{pkg}	260	^oC

NOTES:

1. $T_J = 25^oC$ to 150^oC.

CAUTION: *Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.*

Electrical Specifications $T_C = 25^oC$, Unless Otherwise Specified

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNITS
OFF STATE SPECIFICATIONS						
Drain to Source Breakdown Voltage	BV_{DSS}	$I_D = 250\mu A$, $V_{GS} = 0V$ (Figure 11)	100	-	-	V
Zero Gate Voltage Drain Current	I_{DSS}	$V_{DS} = 95V$, $V_{GS} = 0V$	-	-	1	μA
		$V_{DS} = 90V$, $V_{GS} = 0V$, $T_C = 150^oC$	-	-	250	μA
Gate to Source Leakage Current	I_{GSS}	$V_{GS} = \pm 20V$	-	-	±100	nA
ON STATE SPECIFICATIONS						
Gate to Source Threshold Voltage	$V_{GS(TH)}$	$V_{GS} = V_{DS}$, $I_D = 250\mu A$ (Figure 10)	2	-	4	V
Drain to Source On Resistance	$r_{DS(ON)}$	$I_D = 33A$, $V_{GS} = 10V$ (Figure 9)	-	0.033	0.040	Ω
THERMAL SPECIFICATIONS						
Thermal Resistance Junction to Case	$R_{\theta JC}$	TO-220	-	-	1.25	$^oC/W$
Thermal Resistance Junction to Ambient	$R_{\theta JA}$		-	-	62	$^oC/W$
SWITCHING SPECIFICATIONS ($V_{GS} = 10V$)						
Turn-On Time	t_{ON}	$V_{DD} = 50V$, $I_D = 33A$, $V_{GS} = 10V$, $R_{GS} = 9.1\Omega$ (Figures 18, 19)	-	-	100	ns
Turn-On Delay Time	$t_{d(ON)}$		-	9.5	-	ns
Rise Time	t_r		-	57	-	ns
Turn-Off Delay Time	$t_{d(OFF)}$		-	40	-	ns
Fall Time	t_f		-	55	-	ns
Turn-Off Time	t_{OFF}		-	-	145	ns
GATE CHARGE SPECIFICATIONS						
Total Gate Charge	$Q_{g(TOT)}$	$V_{GS} = 0V$ to 20V; $V_{DD} = 50V$, $I_D = 33A$, $I_{g(REF)} = 1.0mA$ (Figures 13, 16, 17)	-	66	79	nC
Gate Charge at 10V	$Q_{g(10)}$	$V_{GS} = 0V$ to 10V	-	35	42	nC
Threshold Gate Charge	$Q_{g(TH)}$	$V_{GS} = 0V$ to 2V	-	2.4	2.9	nC
Gate to Source Gate Charge	Q_{gs}		-	5.4	-	nC
Gate to Drain "Miller" Charge	Q_{gd}		-	13	-	nC
CAPACITANCE SPECIFICATIONS						
Input Capacitance	C_{ISS}	$V_{DS} = 25V$, $V_{GS} = 0V$, f = 1MHz (Figure 12)	-	1220	-	pF
Output Capacitance	C_{OSS}		-	295	-	pF
Reverse Transfer Capacitance	C_{RSS}		-	100	-	pF

Source to Drain Diode Specifications

PARAMETER	SYMBOL	TEST CONDITIONS	MIN	TYP	MAX	UNITS
Source to Drain Diode Voltage	V_{SD}	$I_{SD} = 33A$	-	-	1.25	V
		$I_{SD} = 17A$	-	-	1.00	V
Reverse Recovery Time	t_{rr}	$I_{SD} = 33A$, $dI_{SD}/dt = 100A/\mu s$	-	-	112	ns
Reverse Recovered Charge	Q_{RR}	$I_{SD} = 33A$, $dI_{SD}/dt = 100A/\mu s$	-	-	400	nC

OPERATING INSTRUCTIONS

INTRODUCTION CODES

Instruction	Set RS	R/W	Instruction Code DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Description	Execution Time (when f_{cp} or f_{osc} is 250 kHz)
Clear Display	0	0	0	0	0	0	0	0	0	1	Clears all display memory and returns the cursor to the home position (Address 0)	82 μs - 1.64ms
Return Home	0	0	0	0	0	0	0	0	1	*	Returns the cursor to the home position (Address 0). Also returns display being shifted to the original position. DD RAM contents remain unchanged.	40 μs - 1.6ms
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S	Sets the cursor move direction and specifies to or not to shift the display. These operations are performed during data write and read.	40 μs - 1.64ms
Display ON/OFF Control	0	0	0	0	0	0	1	D	C	B	D) is display ON/OFF control; memory remains unchanged in OFF condition. (C) cursor ON/OFF (B) blinking cursor.	40 μs
Cursor or Display Shift	0	0	0	0	0	1	S/C	R/L	*	*	Moves the cursor and shifts the display without changing DD RAM contents.	40 μs
Function Set	0	0	0	0	1	DL	N	F	*	*	Sets interface data length (DL), number of display lines (N), and character font (F).	40 μs
Set CG RAM Address	0	0	0	1	A_{CG}						Sets the CG RAM address. CG RAM data is sent and received after this setting.	40 μs
Set DD RAM Address	0	0	1	A_{DD}							Sets the DD RAM address. DD RAM data is sent and received after this setting.	40 μs
Read Busy Flag & Address	0	1	BF	AC							Reads Busy Flag (BF) indicating internal operation is being performed and reads address counter contents.	1 μs
Write Data to CG or to DD RAM	1	0	Write Data								Writes data into DD RAM or CG RAM	40 μs
Read Data from CG or DD RAM	1	1	Read Data								Reads data from DD RAM or CG RAM	40 μs

* Doesn't matter

DD RAM: Display data RAM
CG RAM: Character generator RAM
A_{CG}: CG RAM address
A_{DD}: DD RAM address corresponds to cursor address
A_C: Address counter used for both DD RAM and CG RAM address

I/D = 1: Increment
I/D = 0: Decrement

S = 1: Display shift
S = 0: No display shift

D = 1: Display ON
D = 0: Display OFF

C = 1: Cursor ON
C = 0: Cursor OFF

B = 1: Blink ON
B = 0: Blink OFF

S/C = 1: Display shift
S/C = 0: Cursor movement

BF = 1: Internal operation in progress
BF = 0: Instruction can be accepted

R/L = 1: Right shift
R/L = 0: Left shift

DL = 1: 8 bits
DL = 0: 4 bits

N = 1: 2 lines (LC1671)

F = 0: 5 x 7 dot matrix

Execution times in the above table indicate the minimum values when operating frequency is 250 kHz.

When f_{osc} is 270 kHz: 40μs x 250/250 = 37μs

CHARACTER FONT CODES (5 x 7 DOT MATRIX)

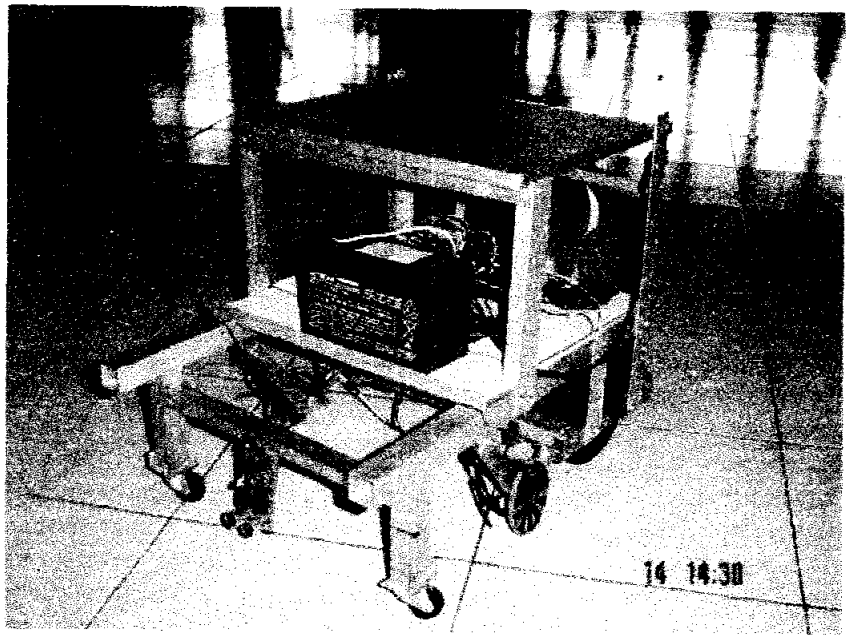
Upper 4 Bit Hexadecimal

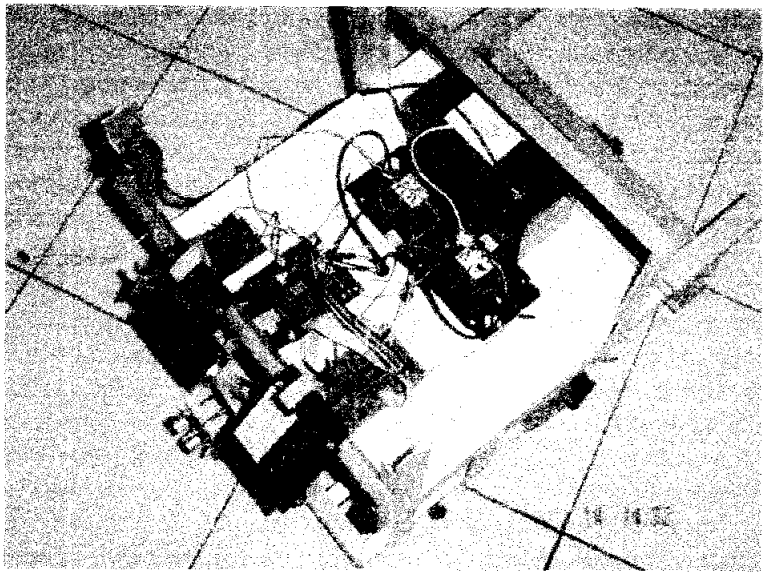
Lower 4 Bit Hexadecimal

Lower 4 bits \ Upper 4 bits	0000 (0)	0010 (2)	0011 (3)	0100 (4)	0101 (5)	0110 (6)	0111 (7)	1010 (A)	1011 (B)	1100 (C)	1101 (D)	1110 (E)	1111 (F)
××××0000 (0)	CG RAM (1)												
××××0001 (1)	(2)												
××××0010 (2)	(3)												
××××0011 (3)	(4)												
××××0100 (4)	(5)												
××××0101 (5)	(6)												
××××0110 (6)	(7)												
××××0111 (7)	(8)												
××××1000 (8)	(1)												
××××1001 (9)	(2)												
××××1010 (A)	(3)												
××××1011 (B)	(4)												
××××1100 (C)	(5)												
××××1101 (D)	(6)												
××××1110 (E)	(7)												
××××1111 (F)	(8)												

FOTO ROBOT UNTUK MENGIKUTI

JALUR YANG TELAH DIAJARKAN





BIODATA

Nama : Fredy

NRP : 5103000075

Tempat, Tgl. Lahir : Surabaya, 10-02-1980

Agama : Kristen

Alamat Rumah : Jl. Kedinding Indah 22,

Surabaya.

Riwayat Pendidikan :

- Tahun 1994 Lulus SDK Pencinta Damai, Surabaya.
- Tahun 1997 Lulus SLTP. St. Stanislaus I, Surabaya.
- Tahun 2000 Lulus SMU. Kr. Petra 3, Surabaya.